HybridAugment++: Unified Frequency Spectra Perturbations for Model Robustness

Mehmet Kerim Yucel\textsuperscript{1} \quad Ramazan Gokberk Cinbis\textsuperscript{2} \quad Pinar Duygulu\textsuperscript{1}
\textsuperscript{1}Hacettepe University, Graduate School of Science and Engineering \\
\textsuperscript{2}Department of Computer Engineering, Middle East Technical University
\texttt{mkerimyucel@gmail.com} \quad \texttt{gcinbis@ceng.metu.edu.tr} \quad \texttt{pinar@cs.hacettepe.edu.tr}

Abstract

Convolutional Neural Networks (CNN) are known to exhibit poor generalization performance under distribution shifts. Their generalization have been studied extensively, and one line of work approaches the problem from a frequency-centric perspective. These studies highlight the fact that humans and CNNs might focus on different frequency components of an image. First, inspired by these observations, we propose a simple yet effective data augmentation method HybridAugment that reduces the reliance of CNNs on high-frequency components, and thus improves their robustness while keeping their clean accuracy high. Second, we propose HybridAugment++, which is a hierarchical augmentation method that attempts to unify various frequency-spectrum augmentations. HybridAugment++ builds on HybridAugment, and also reduces the reliance of CNNs on the amplitude component of images, and promotes phase information instead. This unification results in competitive to or better than state-of-the-art results on clean accuracy (CIFAR-10/100 and ImageNet), corruption benchmarks (ImageNet-C, CIFAR-10-C and CIFAR-100-C), adversarial robustness on CIFAR-10 and out-of-distribution detection on various datasets. HybridAugment and HybridAugment++ are implemented in a few lines of code, does not require extra data, ensemble models or additional networks\textsuperscript{1}.

1. Introduction

The last decade witnessed machine learning (ML) elevating many methods to new heights in various fields. Despite surpassing human performance in multiple tasks, the generalization of these models are hampered by distribution shifts, such as adversarial examples [54], common image corruptions [21] and out-of-distribution samples [62]. Addressing these issues are of paramount importance to facilitate the wide-spread adoption of ML models in practical deployment, especially in safety-critical ones [46, 11], where such distribution shifts are simply inevitable.

Distribution shift-induced performance drops signal a gap between how ML models and us humans perform perception. Several studies attempted to bridge, or at least understand, this gap from architecture [2, 63, 48] and training data [21, 57, 29, 7, 39, 4, 22] centric perspectives. An interesting perspective is built on the frequency spectra of the training data; convolutional neural networks (CNN) are shown to leverage high-frequency components that are invisible to humans [56] and also shown to be reliant on the amplitude component, as opposed to the phase component humans favour [7]. Several studies leveraged frequency spectra insights to improve model robustness. These methods, however, either leverage cumbersome ensemble models [48], formulate complex augmentation regimes [52, 34] or focus on a single robustness venue [33, 52, 34] rather than improving the broader robustness to various distribution shifts. Furthermore, it is imperative to preserve, if not improve, the clean accuracy levels of the model while improving its robustness.

Our work aims to improve the robustness of CNNs to various distribution shifts. Inspired by the frequency spectra based data augmentations, we propose HybridAugment, inspired from the well-known hybrid images [45]. Based on the observation that the label information of images are predominantly related to the low-frequency components [58, 31], HybridAugment simply swaps high-frequency and low-frequency components of randomly selected images in a batch, regardless of their class labels. This forces the network to focus on the low-frequency information of images and makes the models less reliant on the high-frequency information, which are often shown to be the root cause of robustness issues [58]. With virtually no training overhead, HybridAugment improves the corruption robustness while preserving or improving the clean accuracy, and additionally induces adversarial robustness.

\textsuperscript{1}Our code is available at \url{https://github.com/MKYucel/hybrid_augment}. 

5718
Figure 1. An overview of our methods HybridAugment (HA) and HybridAugment++ (HA++), and their single image (S) and paired (P) variants. HA_P combines the high-frequency (HF) and low-frequency (LF) contents of two randomly selected images, whereas HA_P++ combines the HF of one image with the amplitude and LF-phase mixtures of two other images. Single image variants perform the same procedure, but based on different augmented versions of a single image.

Additionally, we set our eyes on jointly exploiting the contributions of frequency spectra augmentation methods while unifying them into a simpler, single augmentation regime. We then propose HybridAugment++, which performs hierarchical perturbations in the frequency spectra. Exploiting the fact that the phase component carries most of the information in an image [7], HybridAugment++ first decomposes images into high and low-frequency components, swaps the amplitude and phase of the low frequency component with another image, and then combines this augmented low-frequency information with the high-frequency component of a random image. Essentially, HybridAugment++ forces the models to rely on the phase and the low-frequency information. As a result, HybridAugment++ further improves adversarial and corruption robustness, while further improving the clean accuracy against several alternatives. See Figure 1 for a diagram of our methods.

Our main contributions can be summarized as follows.

- We propose HybridAugment, a simple data augmentation method that helps models rely on low-frequency components of data samples. It is implemented in just three lines of code and has virtually no overhead.
- We extend HybridAugment and propose HybridAugment++, which performs hierarchical augmentations in frequency spectra to help models rely on low-frequency and phase components of images.
- We show that HybridAugment improves corruption robustness of multiple CNN models, while preserving (or improving) the clean accuracy. We additionally observe clear improvements in adversarial robustness over strong baselines via HybridAugment.
- HybridAugment++ similarly outperforms many alternatives by further improving corruption and clean accuracies on multiple benchmark datasets, with additional gains in adversarial robustness.

2. Related Work

Robust Generalization - Adversarial. Adversarial ML has been studied intensively [54, 66], resulting into numerous attack [54, 41, 14] and defense [36, 50, 3, 35] methods borne out of an arms race that is still very much active. Notable attacks include FGSM [14], DeepFool [41], C&W [5] where AutoAttack [8] is now a widely used attack for adversarial evaluation. The defense methods mainly diversify the training distribution with attacked images [36, 70], purify the adversarial examples [50, 37] or detect whether an image is adversary or not [61, 35].

Robust Generalization - Corruptions. Common image corruptions might have various causes, and they occur more frequently than adversaries in practice. Numerous datasets simulating these effects have been released to fa-
cilitate standard evaluations [21, 42, 24, 67]. The methods addressing corruption robustness can be largely divided into two: architecture-centric and data-centric methods. Architecture-centric methods include neural architecture search for robust architectures [40], focusing on subnets [16], rectifying batch normalization [2], wavelet-based networks [47, 4], learned augmentation policies [64], shape-bias injection [15, 53], style augmentation [13], fractals [22], soft-edge driven image blending [29] and max-entropy image transformations [39] are all shown to improve corruption robustness at varying degrees.

Robust Generalization - Frequency Aspect. Several frequency-centric studies on model generalization show that CNNs tend to rely on high-frequency information ignored by human vision [56], or rely more on amplitude component than phase component humans tend to favour [7]. Models trained on high-pass filtered images are shown to have higher accuracy than the models trained on low-pass filtered images, although high-pass filtered images are just random noise to humans [64]. Multiple studies confirm that models reliant on low-frequency components are more robust [58, 31]. Interestingly, frequency analyses presents a different interpretation of the robustness-accuracy trade-off; many methods that improve clean accuracy force networks to rely on high-frequency components, which might sacrifice robustness [56].

Robust Generalization - Frequency-Centric Methods. A trade-off in frequency-based data augmentations is that one should not sacrifice the other; training on high-frequency augmentations can improve robustness to high-frequency corruptions, but tend to sacrifice the low-frequency corruption robustness or the clean accuracy [48, 64, 6]. Frequency-centric methods include biasing Jacobians [6], swapping phase and amplitude of random images [7], perturbing phase and amplitude spectra along with consistency regularization [52], frequency-band expert ensembles [48], frequency-component swapping of same-class samples [43] and wavelet denoising layers [30]. Note that there is a considerable literature on frequency-centric adversarial attacks, but we primarily focus on methods improving robustness.

A similar work is [43], where hybrid-image based augmentation is proposed. We have, however, several key advantages; we i) lift the restriction of sampling from same classes for augmentation, ii) propose both single and paired variants, leading to a significantly more diverse training distribution, iii) present HybridAugment++ that performs phase/amplitude swap specifically in low-frequency components and iv) report improvements on corruption and adversarial robustness, as well as clean accuracy on multiple benchmark datasets (CIFAR-10/100, ImageNet). Note that other methods either train with ImageNet-C corruptions [48], report only corruption results [52], rely on external data [22] or models [4]. Our methods, on the other hand, require no external models or data, and they can be plugged into existing pipelines easily due to their simplicity.

3. Method

In this section, we formally define the problem, motivate our work and then present our proposed techniques.

3.1. Preliminaries

Let \( \mathcal{F}(x; W) \) be an image classification CNN trained on the training set \( \mathcal{T}_{\text{train}} = (x_i, y_i)_{i=1}^N \) with \( N \) samples, where \( x \) and \( y \) correspond to images and labels. The clean accuracy (CA) of \( \mathcal{F}(x; W) \) is formally defined as its accuracy over a clean test set \( \mathcal{T}_{\text{test}} = (x_j, y_j)_{j=1}^M \). Assume two operators \( \mathcal{A}() \) and \( \mathcal{C}(c, s) \) that adversarially attacks or corrupts a given set of images with the corruption category \( c \) and severity \( s \), respectively. Let \( \mathcal{A}_{\text{test}} \) and \( \mathcal{C}_{\text{test}} \) be the adversarially attacked and corrupted versions of \( \mathcal{T}_{\text{test}} \), and let \( \mathcal{F}(x; W) \) have a robust accuracy (RA) on \( \mathcal{A}_{\text{test}} \) and a corruption accuracy (CRA) on \( \mathcal{C}_{\text{test}} \). The aim is to fit \( \mathcal{F}(x; W) \) such that the model gains robustness (i.e. increased RA and CRA compared its the baseline version), while retaining (or improving) the clean accuracy of its baseline version trained without robustness concerns.

What we know. Our work builds on the following crucial observations: i) CNNs favour high-frequency content [56], ii) adversaries and corruptions often reside in high-frequency [58], iii) images are dominated by low-frequency [48] and iv) models relying on low-frequency components are more robust [31, 58]. The robustness-accuracy trade-off is visible; low-frequency reliant models are more robust, but tend to miss out on clean accuracy brought by the high-frequency components.

3.2. HybridAugment

We hypothesize that a sweet spot in the robustness-accuracy trade-off can be found. Unlike the hard approaches that completely rule out the reliance on high-frequency components (i.e. low-pass filters), we propose to reduce the reliance on them. To this end, we adopt a data augmentation approach that aims to diversify \( \mathcal{T}_{\text{train}} \) by an operation \( \mathcal{HA}() \). Keeping the strong relation intact between labels and low-frequency content (i.e. labels come from low-frequency-component image), we propose to swap high and low-frequency components of images in a batch on-the-fly. Unlike [43], we do not restrict the images to belong to the same class; this diversifies the training distribution even further while preserving the image semantics. We call this basic version of our approach HybridAugment:

\[
\mathcal{H}_A(x_i, x_j) = \mathcal{L}_F(x_i) + \mathcal{H}_F(x_j)
\]  

(1)
where \( x_i \) is the input image and \( x_j \) is a randomly sampled image from the whole training set, which we simply sample from the mini batch at each training iteration in practice. \( \mathcal{H}F \) and \( \mathcal{L}F \) operators select the high and low-frequency components of an input image, for which we use:

\[
\mathcal{L}F(x) = \text{GaussBlur}(x) \\
\mathcal{H}F(x) = x - \mathcal{L}F(x)
\] (2)

where \( \text{GaussBlur} \) is used as a low-pass filter. Note that a similar outcome is possible by using Discrete Fourier Transforms (DFT), swapping the frequency bands and then applying Inverse DFT (IDFT). We find the gaussian blur operation to be faster and better in practice.

Inspired from [7], in addition to the image-pair scheme in Eq. 1, we propose a single image variant of HybridAugment. In the single image variant, instead of combining two images, \( x_i \) and \( x_j \) are obtained by applying randomly sampled augmentations to a single image. The single image variant \( \mathcal{H}A_S \) can therefore be defined as

\[
\mathcal{H}A_S(x_i) = \mathcal{L}F(\text{Aug}(x_i)) + \mathcal{H}F(\hat{\text{Aug}}(x_i))
\] (3)

where \( \text{Aug} \) and \( \hat{\text{Aug}} \) correspond to two sets of randomly sampled augmentation operations. Note that paired and single versions can work in tandem (\( \mathcal{H}APr_S \)), and actually outperform single or paired image versions.

3.3. HybridAugment++

The frequency analysis is a vast literature, however, two core aspects often stand out; frequency-band analysis (i.e. low, high) and the decomposition of signals into amplitude and phase. HybridAugment covers the former and shows competitive results in various benchmarks (see Section 4). The latter is investigated in \( \mathcal{APR} \) [7], where phase is shown to be the more relevant component for correct classification, and training models based on their phase labels and swapping amplitude components of images randomly lead to more robust models. Note that frequency-band and phase/amplitude discussions are arguably orthogonal, since frequency, phase and amplitude provide distinct characterizations of a signal: intuitively speaking, frequency, phase and amplitude can be seen as the separation of visual patterns in terms of scale, location and significance.

We hypothesize these two approaches can be complementary; a model reliant on low-frequency and spatial information (i.e. phase) can further improve robustness. Inspired by the successes of cascaded augmentation methods [21, 57, 4], we unify these two core aspects into a single, hierarchical augmentation method. We refer to this method as HybridAugment++ and define its paired version as:

\[
\mathcal{H}A_{P++}(x_i, x_j, x_z) = \mathcal{APR}_P(\mathcal{L}F(x_i), x_z) + \mathcal{H}F(x_j)
\] (4)

where \( x_i \), \( x_j \) and \( x_z \) are images sampled from the same batch. Here, \( \mathcal{APR}_P \) [7] is defined as

\[
\mathcal{APR}_P(x_i, x_z) = IDFT(A_{x_z} \otimes e^{iP_{x_i}})
\] (5)

where \( \otimes \) is element-wise multiplication, \( A \) is the amplitude and \( P \) is the phase component. Similar to \( \mathcal{H}A \) and \( \mathcal{APR} \), we also define a single-image version of HybridAugment++ as

\[
\mathcal{H}A_{S++}(x_i) = \mathcal{APR}_S(\mathcal{L}F(\text{Aug}(x_i))) + \mathcal{H}F(\hat{\text{Aug}}(x_i))
\] (6)

where \( \mathcal{APR}_S \) [7] is defined as

\[
\mathcal{APR}_S(x_i) = IDFT(A_{\text{Aug}(x_i)} \otimes e^{iP_{\text{Aug}}(x_i)})
\] (7)

where \( \text{Aug} \), \( \hat{\text{Aug}} \), \( \text{Aug} \) and \( \hat{\text{Aug}} \) are different sets of randomly sampled augmentation operations. Note that we essentially propose a framework; one can use different single and paired image augmentations, either individually or together, and can still achieve competitive results (see ablations in Section 4). There are also other alternatives, such as swapping phase/amplitude first and then performing \( \mathcal{H}A \), but we observe poor performance in practice; dividing the phase component into frequency-bands is not interpretable as frequencies of the phase component are not well defined. The pseudo-code of our methods can be found in the supplementary material.

4. Experimental Results

In this section, we first describe our experimental setup, including the datasets, metrics, architectures and implementation details. We then present a discussion of the Gaussian kernel details, an important detail of the proposed schemes. We thoroughly evaluate the effectiveness of \( \mathcal{H}A \) and \( \mathcal{H}A^{++} \) in terms of three distribution shifts; common image corruptions, adversarial attacks, and out-of-distribution detection. We finalize with additional results and a discussion of the potential limitations.

4.1. Experimental setup

Datasets. We use CIFAR-10, CIFAR-100 [27] and ImageNet [10] for training. Both CIFAR datasets are formed of 50,000 training images with a size of 32x32. ImageNet dataset contains around 1.2 million images of 1000 different classes. Corruption robustness is evaluated on the corrupted versions of the test splits of these datasets, which are CIFAR-10-C, CIFAR-100-C and ImageNet-C [19]. For each dataset, corruptions are simulated for 4 categories (noise, blur, weather, digital) with 15 corruption types, each with 5 severity levels. For adversarial robustness, we use AutoAttack [8] on CIFAR-10 test set. Out-of-distribution detection is evaluated on SVHN [44], LSUN [65], ImageNet and CIFAR-100, and their fixed versions [55].
Evaluation metrics. We report top-1 classification as clean accuracy. Adversarial robustness is evaluated with robust accuracy, which is the top-1 classification on adversarially attacked test sets. Corruption robustness is evaluated with Corruption Error (CE) $CE = \sum_{c,s} E_{c,s}^{F} / \sum_{c,s} E_{c,s}^{AlexNet}$. CE calculates the average error of the model $F$ on a corruption type, normalized by the corruption error of AlexNet [28]. CE is calculated for all 15 corruption types, and their average Mean Corruption Error (mCE) is used as the final robustness metric. Out-of-distribution detection is evaluated using the Area Under the Receiver Operating Characteristic Curve (AUROC) metric [20].

Architectures. We use architectures commonly used in the literature for a fair comparison; ResNeXt [60], All-Convolutional Network [51], DenseNet [23], WideResNet [68] and ResNet18 [17] are used in CIFAR-10 and CIFAR-100, whereas ResNet50 is used for ImageNet.

Implementation details. For CIFAR experiments, all architectures are trained for 200 epochs with SGD, where initial learning rate of 0.1 decays after every 60 epochs. We use the last checkpoints for evaluation and do not perform any hyperparameter tuning. Paired and single variants of $H_A$ and $H_A^{++}$ are applied in each iteration with probabilities 0.6 and 0.5, respectively. Standard training augmentations are random horizontal flips and cropping. When a single-image augmentation is used, the input image is augmented with Aug randomly sampled among [rasterize, autocorrupt, equalize, rotate, solarize, shear, translate]. Note that these do not overlap with test corruptions. On ImageNet, we train for 100 epochs with SGD, where an initial learning rate of 0.1 is decayed every 30 epochs. Data augmentations and their probabilities are the same as above.

We use the same checkpoints for all evaluations; we do not train separate models for corruption and out-of-distribution detection. In adversarial analysis, for a fair comparison with [7], we train our model with $\mathcal{H}_A \& \mathcal{H}_A^{++}$ and FGSM adversarial training. We note that we use the labels of the low-frequency image as the ground-truth labels. We have tried using the high-frequency image labels instead, but this leads to severe degradation in overall performance, as expected. All models are trained with the cross-entropy loss, where the original and the augmented (with our method) batches are used to calculate the loss.

4.2. Understanding the cut-off frequency

A key design choice is the cut-off frequency that defines $\mathcal{H}_F$ and $\mathcal{L}_F$ in Equation 2. Since we essentially define the cut-off frequency with a Gaussian blur operation, we have two hyperparameters; the size of the Gaussian kernel $K$ and its standard deviation $S$. Note that increasing both the kernel size and the standard deviation increases the blur strength, which eliminates increasingly higher frequencies (i.e., higher cut-off frequency). We now evaluate the effects of these hyperparameters on both clean accuracy and mean corruption errors using $\mathcal{H}_A^{++}$, on both CIFAR-10 and CIFAR-100 using the ResNet18 architecture.

Fixed standard deviation. The effect of different $K$ values with fixed $S = 0.5$ is shown in Figure 2 top plot. $K = 3$ provides the best trade-off here; it has the best clean accuracy and mCE on CIFAR100, whereas it shares the best mCE and has competitive clean accuracy on CIFAR10.

Fixed kernel size. $K = 3$ with different standard deviation values are shown in Figure 2 bottom plot. The robustness-accuracy trade-off becomes more visible here; lower sigma values (i.e., lower cut-off frequency) preserve more high-frequency content, and therefore have increasingly higher clean accuracy, but at the expense of degrading mCE. Note that further increasing the value $S$ is in contrast with this phenomena; if our method had only done frequency swapping (i.e., $H_A$), then we could have expected a consistent trend, as shown in the literature [31, 58]. However, $H_A^{++}$ also emphasizes the phase components, which results into a favourable behaviour where best results in mCE and clean accuracy can be obtained in the same cut-off frequency.

The takeaway. The results show that our hypothesis holds; we can find a sweet spot in the frequency spectrum where we can obtain favourable performance on both corrupted
and clean images, given a careful selection of $K$ and $S$. A sound argument is that the optimality of these hyperparameters depends on the data; this is probably a correct assumption and can help tune the results further on other datasets. However, we use $K = 3$, $S = 0.5$ on all experiments across all architectures and datasets (including ImageNet), and show that we get solid improvements without any dataset-specific tuning.

4.3. Corruption robustness

As mentioned in Section 3, we have three augmentation options; APR [7], $\mathcal{H}_{A}$ and $\mathcal{H}_{A}^{++}$. We can apply them using image pairs, a single image or we can do both. This leads to quite a few potential combinations. We now evaluate all these combinations on CIFAR-10 and CIFAR-100, both for clean accuracy and corruption robustness (mCE).

Comparison against RFC [43]. We implement and compare against RFC, which also performs hybrid-image based augmentation. RFC operates on paired-images of same-class samples, therefore we first compare it against $\mathcal{H}_{A_{P}}$ and $\mathcal{H}_{A_{P}^{++}}$. In mCE, we comfortably outperform it while staying competitive in clean accuracy. This shows the value of lifting the limitation of class-based sampling, which RFC does. Note that since we also propose single-image variants, both single-image augmentations ($\mathcal{H}_{A_{S}}$ and $\mathcal{H}_{A_{S}^{++}}$) and combined ones ($\mathcal{H}_{A_{PS}}$ and $\mathcal{H}_{A_{PS}^{++}}$) significantly outperform RFC on all architectures, datasets and metrics.

Corruption robustness. The corruption results are shown in Table 1. The take-away message is crystal clear; $\mathcal{H}_{A}^{++}$ is the best on all datasets, all architectures and all groups. The best results are obtained when we use $\mathcal{H}_{A}^{++}$ both in pairs and single images, further cementing its effectiveness. Note that $\mathcal{H}_{A}$ is competitive or better than APR.

Clean Accuracy. The clean accuracy values of the models shown in Table 1 are given in Table 2. The results show us that both $\mathcal{H}_{A}$ and $\mathcal{H}_{A}^{++}$ achieve a good spot in robustness-accuracy trade-off; except two cases, both of them improve clean accuracy over the original models. The results are not as clean-cut as those of Table 1, but in each group, the best ones mostly include $\mathcal{H}_{A}$ or $\mathcal{H}_{A}^{++}$. Furthermore, the best results on CIFAR-10 and CIFAR-100 have $\mathcal{H}_{A_{S}}$ as the single-image augmentation. Although it does not perform the best, $\mathcal{H}_{A_{PS}^{++}}$ still outperforms the baseline and is highly competitive against others.

The takeaway. The results show us that $\mathcal{H}_{A}$ and $\mathcal{H}_{A}^{++}$ are superior to other frequency-based methods, and they comfortably improve robustness and clean accuracy performance across multiple datasets and architectures. See supplementary material for comparison with the state-of-the-art on CIFAR-10 and CIFAR-100. Hint to readers: we achieve the state-of-the-art on all architectures on both datasets.

4.3.1 Scaling to ImageNet

We now assess whether our methods can scale to ImageNet. Since we do not use extra data or ensembles during training or inference, we choose methods with similar characteristics, such as SIN [47], PatchUniform, AutoAugment (AA), Random AA [9], MaxBlurPool and AugMix [21]. The results are shown in Table 3. Note that we use pretrained weights for alternative methods if available, otherwise we use the values reported in [7].

The results show that all of our variants produce higher clean accuracy compared to APR, showing the value of our method in improving model accuracy. $\mathcal{H}_{A}$ results are competitive in corruption accuracy, but $\mathcal{H}_{A}^{++}$ outperforms both APR and others in corruption accuracy, while being 0.5 shy of our best clean accuracy. Furthermore, our approach works well with extra data and other augmentations; we apply $\mathcal{H}_{A_{PS}^{++}}$ with DeepAugment [18] and AugMix [21], which leads to significant improvements in mCE (∼11 points) over both DeepAugment and $\mathcal{H}_{A_{PS}^{++}}$. Note that we are better than APR, even when both methods train with DeepAugment. We also outperform PixMix [22], which uses extra training data. Finally, we provide results of $\mathcal{H}_{A_{PS}^{++}}$ with higher cut-off frequency (see experiments with † in Table 3); we see the expected trend where the elimination of higher frequencies make our models more robust in average, at the expense of lowered clean accuracy.

Qualitative results. We provide GradCam visualizations of $\mathcal{H}_{A}^{++}$ against various corruptions in Figure 3. We sample corruptions from each category; noise, motion blur, fog,
scores are the best results within their respective group (i.e. single-only, paired-only, etc.). The overall best results are shown in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Orig</th>
<th>Single-only</th>
<th>RFC(43)</th>
<th>Paired-only</th>
<th>APR$_{P[7]}$</th>
<th>APR$_{S}$</th>
<th>H$_A$</th>
<th>H$_{A+}$</th>
<th>H$_{A+}^{++}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllConv</td>
<td>30.3</td>
<td>14.9</td>
<td>16.8</td>
<td>15.9</td>
<td>16.1</td>
<td>16.2</td>
<td>11.5</td>
<td>11.9</td>
<td>11.2</td>
</tr>
<tr>
<td>DenseNet</td>
<td>30.7</td>
<td>12.3</td>
<td>15.0</td>
<td>11.1</td>
<td>20.4</td>
<td>20.3</td>
<td>18.4</td>
<td>14.2</td>
<td>14.3</td>
</tr>
<tr>
<td>WRResNet</td>
<td>26.9</td>
<td>10.6</td>
<td>13.6</td>
<td>10.0</td>
<td>18.3</td>
<td>18.3</td>
<td>16.4</td>
<td>13.2</td>
<td>13.2</td>
</tr>
<tr>
<td>ResNeXt</td>
<td>27.5</td>
<td>11.0</td>
<td>13.2</td>
<td>9.99</td>
<td>19.2</td>
<td>18.5</td>
<td>17.6</td>
<td>13.2</td>
<td>13.2</td>
</tr>
<tr>
<td>ResNet18</td>
<td>25.4</td>
<td>9.9</td>
<td>12.2</td>
<td>9.34</td>
<td>19.6</td>
<td>17.0</td>
<td>18.3</td>
<td>15.2</td>
<td>15.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Orig</th>
<th>Single-only</th>
<th>RFC(43)</th>
<th>Paired-only</th>
<th>APR$_{P[7]}$</th>
<th>APR$_{S}$</th>
<th>H$_A$</th>
<th>H$_{A+}$</th>
<th>H$_{A+}^{++}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllConv</td>
<td>56.4</td>
<td>39.8</td>
<td>43.0</td>
<td>38.9</td>
<td>50.8</td>
<td>47.5</td>
<td>44.7</td>
<td>41.7</td>
<td>41.7</td>
</tr>
<tr>
<td>DenseNet</td>
<td>59.3</td>
<td>38.3</td>
<td>41.3</td>
<td>37.3</td>
<td>52.5</td>
<td>49.8</td>
<td>45.6</td>
<td>41.8</td>
<td>41.8</td>
</tr>
<tr>
<td>WRResNet</td>
<td>55.3</td>
<td>35.5</td>
<td>38.1</td>
<td>33.9</td>
<td>47.4</td>
<td>44.7</td>
<td>43.1</td>
<td>39.3</td>
<td>39.3</td>
</tr>
<tr>
<td>ResNeXt</td>
<td>53.4</td>
<td>33.7</td>
<td>35.6</td>
<td>31.1</td>
<td>46.4</td>
<td>44.2</td>
<td>41.2</td>
<td>36.4</td>
<td>36.4</td>
</tr>
<tr>
<td>ResNet18</td>
<td>51.2</td>
<td>33.0</td>
<td>35.6</td>
<td>32.1</td>
<td>47.5</td>
<td>49.2</td>
<td>45.5</td>
<td>44.6</td>
<td>44.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Orig</th>
<th>Single-only</th>
<th>RFC(43)</th>
<th>Paired-only</th>
<th>APR$_{P[7]}$</th>
<th>APR$_{S}$</th>
<th>H$_A$</th>
<th>H$_{A+}$</th>
<th>H$_{A+}^{++}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllConv</td>
<td>93.3</td>
<td>93.5</td>
<td>94.1</td>
<td>93.9</td>
<td>93.6</td>
<td>93.1</td>
<td>93.3</td>
<td>93.3</td>
<td>93.3</td>
</tr>
<tr>
<td>DenseNet</td>
<td>94.2</td>
<td>94.9</td>
<td>95.3</td>
<td>95.0</td>
<td>93.6</td>
<td>92.1</td>
<td>90.3</td>
<td>93.3</td>
<td>93.3</td>
</tr>
<tr>
<td>WRResNet</td>
<td>94.8</td>
<td>95.0</td>
<td>95.3</td>
<td>95.4</td>
<td>94.7</td>
<td>93.2</td>
<td>92.2</td>
<td>92.2</td>
<td>92.2</td>
</tr>
<tr>
<td>ResNeXt</td>
<td>95.7</td>
<td>95.5</td>
<td>95.3</td>
<td>95.7</td>
<td>93.5</td>
<td>95.5</td>
<td>93.5</td>
<td>92.9</td>
<td>92.9</td>
</tr>
<tr>
<td>ResNet18</td>
<td>92.2</td>
<td>95.6</td>
<td>95.5</td>
<td>95.6</td>
<td>91.7</td>
<td>94.0</td>
<td>90.9</td>
<td>89.7</td>
<td>89.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Orig</th>
<th>Single-only</th>
<th>RFC(43)</th>
<th>Paired-only</th>
<th>APR$_{P[7]}$</th>
<th>APR$_{S}$</th>
<th>H$_A$</th>
<th>H$_{A+}$</th>
<th>H$_{A+}^{++}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllConv</td>
<td>94.2</td>
<td>94.9</td>
<td>94.9</td>
<td>94.9</td>
<td>93.0</td>
<td>95.0</td>
<td>92.9</td>
<td>92.3</td>
<td>92.3</td>
</tr>
<tr>
<td>DenseNet</td>
<td>97.4</td>
<td>96.3</td>
<td>96.6</td>
<td>96.9</td>
<td>96.4</td>
<td>96.7</td>
<td>96.0</td>
<td>95.9</td>
<td>95.9</td>
</tr>
<tr>
<td>WRResNet</td>
<td>71.4</td>
<td>75.8</td>
<td>76.0</td>
<td>75.6</td>
<td>71.6</td>
<td>71.5</td>
<td>71.4</td>
<td>71.7</td>
<td>71.7</td>
</tr>
<tr>
<td>ResNeXt</td>
<td>72.1</td>
<td>76.2</td>
<td>76.6</td>
<td>76.2</td>
<td>72.1</td>
<td>70.4</td>
<td>71.3</td>
<td>71.7</td>
<td>71.7</td>
</tr>
<tr>
<td>ResNet18</td>
<td>75.0</td>
<td>77.9</td>
<td>79.4</td>
<td>79.4</td>
<td>74.2</td>
<td>71.1</td>
<td>73.5</td>
<td>74.3</td>
<td>74.3</td>
</tr>
<tr>
<td>Mean</td>
<td>72.9</td>
<td>76.6</td>
<td>76.9</td>
<td>76.8</td>
<td>71.9</td>
<td>70.3</td>
<td>71.1</td>
<td>70.8</td>
<td>70.8</td>
</tr>
</tbody>
</table>

Table 1. Corruption robustness on CIFAR-10 (first 6 rows) and CIFAR-100 with various CNNs. Values show mCE, lower is better. Underlined scores are the best results within their respective group (i.e. single-only, paired-only, etc.). The overall best results are shown in bold. The table is divided into groups for easy comparison; single-only augmentation, paired-only augmentation and fixing one augmentation in paired variants while changing the single-image augmentation.

Table 2. Clean accuracy values on CIFAR-10 (first 6 rows) and CIFAR-100. Higher the better. Underlined scores are the best results within their respective group (i.e. single-only, paired-only, etc.). The overall best results are shown in bold.

4.4. Adversarial Robustness

We present our results on adversarial robustness in Table 4. For a fair comparison, we train models from scratch if official code is available. If not, we use pretrained models or use the results reported in [7]. We compare against APR, Cutout and FGSM adversarial training [36].

Our results show that there is no clear winner; with H$_A$, we obtain the best clean accuracy and with H$_{A+}^{++}$ we obtain the best robust accuracy. All our variants are better than the widely accepted adversarial training (AT) baseline in nearly all cases, which shows the effectiveness of our method. Our variants do quite well in clean accuracy and outperform others in nearly all cases. H$_{A+}^{++}$ offers arguably the best trade-off; it ties with APR$_{P[S]}$ on robust accuracy, and outperforms it on clean accuracy.

4.5. Out-of-Distribution Detection

For OOD detection, we use a ResNet18 model trained on CIFAR-10 and compare against several configurations, such as training with cross-entropy, SupCLR [25] and CSI [55], and augmentation methods as Cutout, Mixup and APR.

First of all, all our variants comfortably beat the baseline OOD detection (CE), which shows that our proposed method is indeed useful. Furthermore, we see that our proposed methods are highly competitive, and they perform as good as the alternative methods. H$_{A+}^{++}$ + APR$_{S}$ outperforms all other methods on LSUN and ImageNet datasets, and produces competitive results on others. Mean AUROC across all datasets show that it ties with the best model APR$_{P[S]}$, showing its efficiency. The broader framework we propose leads to many variants with various performance profiles across different datasets, highlighting the flexibility and usefulness of our unification of frequency-centric augmentations. Note that the clean accuracy on CIFAR-10 are provided in Table 2, and shows that we perform the same or better than the other methods.
Table 3. Clean error and corruption robustness on ImageNet. Lower is better. The methods shown in the last four rows leverage extra data during training. † indicates training with a higher cut-off frequency.

Table 4. Clean and robust accuracy (CA, RA) on CIFAR-10 attacked with AutoAttack [8]. Higher the better.

Table 5. Out-of-distribution AUROC results on multiple datasets. Higher the better. Our models are trained with CE as well. † indicates fixed versions of respective datasets. CIF100 is CIFAR100.

4.6. Additional results and potential limitations

ImageNet-Ç. We also assess our models on ImageNet-Ç [38]. The results, given in Table 6, show key insights: we significantly improve over the original standard model and we are just 0.1 shy of $AP_{\mathcal{PS}}$. Training with additional data [18] helps, and actually puts us ahead of $AP_{\mathcal{PS}}$. An interesting observation is that with higher cut-off frequency (i.e. stronger blur), the performance becomes worse; in ImageNet-Ç, we observe the opposite. This is potentially due to the different dominant frequency bands in corruptions of ImageNet-Ç and ImageNet-Ç.

What about transformers? We also train a Swin-Tiny [32] on ImageNet with and without $\mathcal{HA}_{++}$; ImageNet-Ç results show improvements (59.5 vs 54.8 mCE), but at the expense of slight degradation on clean accuracy (81.2 vs 80.6 top-1). Despite the fundamental differences between transformers and CNNs, especially regarding the frequency bands of the features they tend to learn [1], it is encouraging to see our methods also work well for transformers. We leave further analyses on transformers for future work.

5. Conclusion

In this paper, inspired by the frequency-centric explanation of how CNNs generalize, we propose two augmentation methods HybridAugment and HybridAugment++. The former aims to reduce the reliance of CNN generalization on high-frequency information in images, whereas the latter does the same but also promotes the use of phase information rather than the amplitude component. This unification of two distinct frequency-based analyses into a data augmentation method leads to results competitive to or better than state-of-the-art on clean accuracy, corruption and adversarial performance and out-of-distribution detection.

Acknowledgements. This work was supported in part by a Google Faculty Research Award.

References

[1] Philipp Benz, Chaoming Zhang, Soomin Ham, Adil Karjaev, and In So Kweon. Robustness comparison of vision trans-


[53] Mingjie Sun, Zichao Li, Chaowei Xiao, Haonan Qiu, Bhavya Kailkhura, Mingyan Liu, and Bo Li. Can shape structure


